

Description du système ITS-2
Composantes de Transfert et de Génération

Mira Ramluckun
Département de linguistique - LATL
Université de Genève

Avril 1994

TABLE DES MATIERES

1	Introduction	2
1.1	Généralités	2
1.2	Structure profonde et de surface	2
1.3	Algorithme de transfert d'une phrase	3
1.3.1	Création du syntagme cible CP - Procédure TranslateCP	3
1.3.2	Recherche du verbe principal - Procédure GetMainVerb	4
1.3.3	Transfert de la racine du verbe principal - Procédure TranslateVerbHead	4
1.3.4	Transfert des arguments sources - Procédure TrArguments	5
1.3.5	Application des transformations - Procédure ApplyTransf	5
1.3.6	Transfert du spécificateur et complément - Procédure TraverseVerb . . .	6
1.3.7	Génération morphologique - Procédure ApplyMorpho	6
1.3.8	Génération de la phrase cible - Procédure StandardWriteList	7
2	Traitement des causatives français/anglais	8
2.1	Algorithme de transfert	8
2.1.1	Distinction entre <i>faire_a</i> et <i>faire_{par}</i> - Procédure SpecifyRestructLex	8
2.1.2	Détermination de la structure profonde du verbe infinitif cible	9
2.1.3	Création de la structure bi-propositionnelle - Procédure TrRestructVerb .	11

1 Introduction

Ce document présente une esquisse des stratégies et algorithmes adoptés dans le système ITS-2 (Interactive Translation System). Il reprend et complète les notes techniques antérieures de ITS-2, entre autres les documents Wehrli, notes techniques 93/2 et Pinnagoda, notes techniques 93/16. Pour une meilleure compréhension du document, il est vivement conseillé d'étudier au préalable les structures des données utilisées par les lexiques et les analyseurs et de se familiariser avec les arbres syntaxiques générés par les analyseurs.

1.1 Généralités

Les composantes de transfert et de génération de ITS-2 ont pour but de dériver à partir d'une structure syntaxique source générée par l'analyseur, la structure de surface cible qui engendre la phrase cible. Ce processus est déterministe. A une structure source correspond une structure cible. Toute ambiguïté de transfert est gérée soit par l'interaction soit par des valeurs par défaut.

L'algorithme général de transfert parcourt et traite récursivement l'arborescence syntaxique en pré-ordre. En d'autres termes, la racine de l'arbre - soit une tête lexicale soit une liste d'éléments coordonnés - est transférée suivie de l'arbre de gauche - le spécificateur - et de l'arbre de droite - le complément.

Lorsqu'une phrase est transférée, la stratégie consiste à créer dans un premier temps, une structure profonde cible sous forme prédicative. Plus précisément, cette structure est représentée par un prédicat verbal, en l'occurrence le verbe principal de la phrase, et les différents arguments qui jouent les rôles grammaticaux tels que sujet, objet direct, objet prépositionnel et ainsi de suite. Des règles transformationnelles et morphologiques sont ensuite mises en oeuvre pour déterminer la structure de surface cible à partir de la structure profonde.

Ainsi lors de la traduction d'une phrase, l'algorithme général décrit ci-dessus est modifié lorsqu'un syntagme verbal est transféré. Le changement principal réside dans la création d'une structure profonde et l'application des règles transformationnelles avant le transfert des arbres gauche et droit du syntagme verbal.

1.2 Structure profonde et de surface

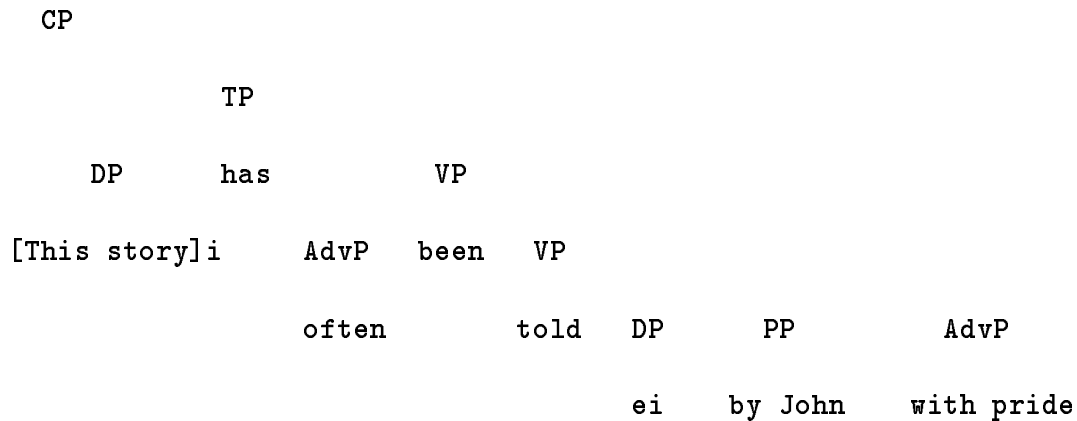
Avant de décrire l'algorithme de transfert, nous allons expliciter les notions de structure profonde et de surface. Prenons la phrase passive (1a)

(1)a. This story has often been told by John with pride.

La structure profonde de cette phrase correspond à la forme active

Tell (argument sujet [DP John], argument objet direct [DP this story])

Quant à la structure de surface, elle est la structure syntaxique suivante :

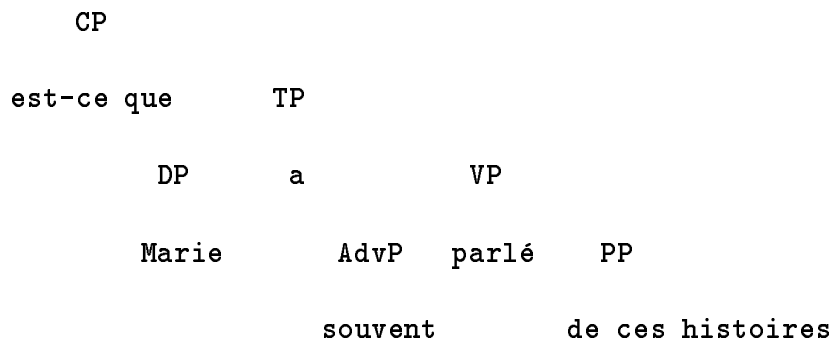


La règle transformationnelle du passif permet de convertir la structure profonde en une structure de surface et inversement. Par ailleurs, notons également que les ajouts adverbiaux [AdvP often] et [AdvP with pride] ne font pas partie de la structure profonde.

1.3 Algorithme de transfert d'une phrase

Les étapes suivantes sont effectuées successivement dans l'algorithme de transfert d'une phrase et seront illustrées par l'exemple (2a)

(2)a. est-ce que Marie a souvent parlé de ces histoires?



1.3.1 Création du syntagme cible CP - Procédure TranslateCP

La racine d'un syntagme CP est en général construite comme un élément vide ayant les traits lexicaux sources. Les seuls cas où la racine du CP est effectivement transférée se produisent

lorsque le syntagme CP est un ajout ou une proposition subordonnée. Dans notre exemple, nous allons donc construire un syntagme CP cible avec une tête vide.

1.3.2 Recherche du verbe principal - Procédure GetMainVerb

Le verbe principal est le dernier syntagme VP de la phrase. Ce verbe exprime la structure profonde source à travers sa table d'arguments. Ainsi, dans notre exemple, cette structure est :

Parler (sujet [DP Marie], objet prépositionnel [PP de ces histoires])

1.3.3 Transfert de la racine du verbe principal - Procédure TranslateVerbHead

Dans une phrase sans coordination des verbes, la racine du verbe principal est une tête lexicale liée à un ou plusieurs lexèmes. Le dictionnaire bilingue fournit la correspondance cible d'un lexème verbal source ainsi que les liens qui existent entre les arguments de ces deux lexèmes. Par conséquent, pour éviter la génération de plusieurs structures cibles, il est nécessaire d'identifier le lexème source qui représente au mieux la structure profonde de la phrase source - Procédure MatchWithLexeme. Au passage, on peut noter que l'analyseur effectue déjà un filtrage des lexèmes. Cependant ce filtrage n'aboutit pas forcément à la sélection d'un lexème.

Dans le dictionnaire français, le verbe *parler* a actuellement quatre lectures de lexèmes :

1. [Jean] parle [PP à Marie].
[DP_sujet] [PP à]
2. [Jean] parle [de la politique] [à Marie]
[DP_sujet] [PP à] [PP de]
3. [Jean] parle [CP d'acheter une maison].
[DP_sujet] [S S_infinitif]
4. [Jean] parle [PP de la politique]
[DP_sujet] [PP de]

Le lexème qui sera sélectionné dans notre exemple est bien entendu la quatrième lecture [DP sujet] [PP de] où [DP sujet] correspond au syntagme [DP Jean] et [PP de] au syntagme [PP de ces histoires].

La correspondance de ce lexème dans le dictionnaire bilingue est

```
parler [DP_sujet] [PP de]
talk   [DP_sujet] [PP of]
```

Avec ces informations, il est aisé de construire la structure profonde cible. En effet, on déduit facilement le tableau suivant - Procédure `MapTargetLexeme` - qui exprime la fonction grammaticale cible et les traits lexicaux cibles associés à un argument source.

Procédure MatchWithLexeme	Procédure MapTargetLexeme
Lexème Source PARLER	Argument Source
Sujet	[DP Marie]
Objet Prép + de	[PP de ces histoires]
	Lexème Cible TALK
	Sujet
	Objet Prép + of
----- Informations du dictionnaire bilingue -----	

1.3.4 Transfert des arguments sources - Procédure `TrArguments`

Chaque argument source est traduit selon l'algorithme de transfert pré-ordre pour créer les arguments cibles de la structure profonde.

Talk (sujet [DP Mary], objet prépositionnel [PP of this story])

1.3.5 Application des transformations - Procédure `ApplyTransf`

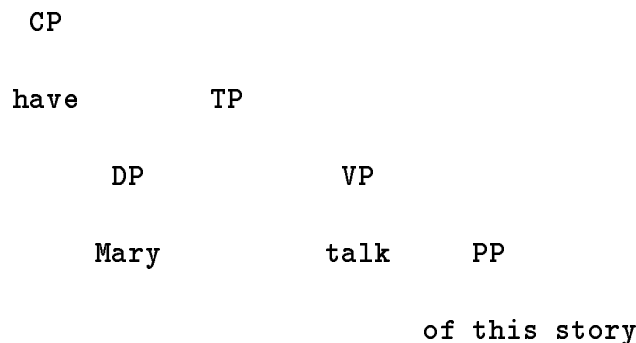
Les règles transformationnelles sont appliquées successivement et s'il y lieu sur la structure profonde pour générer la structure de surface. Ces règles concernent :

- le passif
- le transfert du temps avec l'insertion des auxiliaires
- les constructions infinitives avec les phrases à montée ou à contrôle
- l'insertion du *do* dans les phrases négatives ou interrogatives
- l'inversion du sujet et de l'auxiliaire dans des phrases interrogatives.

Toutes ces règles transformationnelles sont décrites dans les documents Wehrli, notes techniques 93/2 et Pinnagoda, notes techniques 93/16. Dans notre exemple, le verbe *parler* est au passé composé et prend conséquemment la forme *Present Perfect - have talked* en anglais. L'auxiliaire *have* est rajouté. D'autre part, la phrase étant interrogative, la transformation d'inversion du sujet et de l'auxiliaire est effectuée. Cette transformation déplace l'auxiliaire *have* dans la position tête du CP.

L'argument sujet de la structure profonde est par la suite inséré dans le spécificateur du premier syntagme verbal. Tous les autres arguments sont placés, selon un ordre inhérent au langage cible, dans le complément du dernier syntagme verbal. En anglais, l'argument objet indirect est le premier inséré dans le complément, suivi successivement de l'objet direct, l'objet prépositionnel, l'argument prédicatif - syntagme FP, et l'argument phrase - syntagme CP.

On aboutit ainsi à la structure :



1.3.6 Transfert du spécificateur et complément - Procédure TraverseVerb

Le spécificateur - arbre gauche - et le complément - arbre droit - du verbe principal sont parcourus afin de transférer les syntagmes, notamment les ajouts, qui ne sont pas encore traités. Ces parcours ne peuvent intervenir qu'après l'application des règles transformationnelles puisque que le placement des ajouts dépend de la structure de surface.

Dans notre exemple. lors du parcours du spécificateur du verbe *parler*, l'ajout [AdvP souvent] est transféré. Lors du parcours du complément, le syntagme [PP de ces histoires] n'est pas transféré de nouveau puisqu'il a déjà été transféré en tant qu'argument.

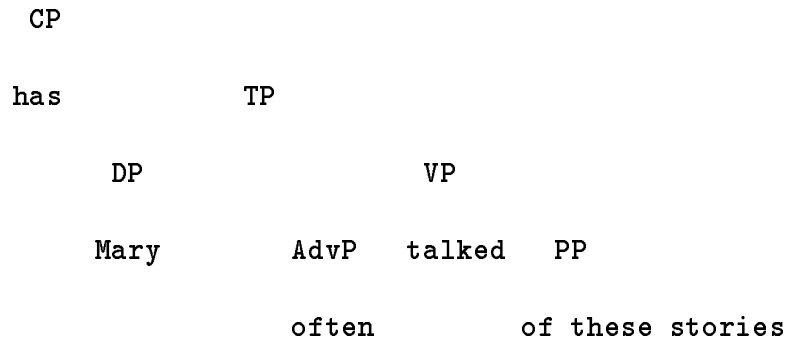
1.3.7 Génération morphologique - Procédure ApplyMorpho

Pendant tout le processus de transfert, la tête lexicale du syntagme est toujours mise en correspondance avec la forme de base de l'item lexical cible. Ce sont uniquement les traits morphologiques (nombre, genre, personne, temps) associés au syntagme cible qui sont mis à jour.

Ainsi, dans notre exemple, le syntagme [NP histoires] est transféré comme [NP story] avec le trait pluriel.

Pendant la phase de génération morphologique, l'arbre syntaxique cible est parcouru en pré-ordre et tout syntagme, qui doit avoir une forme morphologique autre que la forme de base, est mis à jour.

On aboutit à l'arbre syntaxique cible :



1.3.8 Génération de la phrase cible - Procédure StandardWriteList

Dans la dernière étape de génération la phrase cible, l'arbre syntaxique est parcouru en-ordre pour afficher la racine de tout arbre. L'ordre de parcours, est le spécificateur - arbre gauche, suivi de la racine et du complément - arbre droit.

Dans notre exemple, la phrase cible générée est (3a)

(3)a. Has Mary often talked of these stories?

2 Traitement des causatives français/anglais

La spécificité du transfert des causatives est dûe au fait que les structures syntaxiques source et cible ne sont pas identiques. En effet, en français, la structure est mono-propositionnelle alors qu'en anglais, elle est bi-propositionnelle.

Le transfert des causatives est basé sur le document Wehrli, notes techniques 93/6. Rappelons simplement ici, qu'une construction causative prend en français la forme d'un verbe opérateur (*faire, laisser, voir, entendre, etc...*) immédiatement suivi d'un verbe infinitif.

Rappelons également que, dans le cas particulier du verbe causative *faire*, on peut distinguer deux types de construction : *faire_à* ou *faire_{par}*. On verra par la suite que la construction *faire_à*, illustrée en (4a), donne lieu en anglais à des structures de type **make** + *V_{inf}*. Quant à la construction *faire_{par}*, elle sera traduite par une structure **have V-en**, comme en (4b).

- (4)a. Jean fait venir ses amis → Jean makes his friends come.
 b. Jean fait réparer sa voiture → Jean has his car fixed.

2.1 Algorithme de transfert

L'algorithme de transfert des causatives reprend l'algorithme de transfert des phrases décrit ci-dessus avec les quelques modifications que nous allons présenter maintenant. L'algorithme est illustré à travers les exemples (5a) pour les constructions *faire_à*, (5b) pour les constructions *faire_{par}* et (5c) pour des constructions avec des verbes autres que *faire*.

- (5)a. Marie fait goûter le vin à un ami.
 b. Jean fait réparer la voiture par un ami.
 c. Paul laisse dormir les enfants.

Une structure causative française est identifiée par la présence du trait **restructuration** dans le champ **v_features** du verbe principal - le verbe infinitif. Ce trait indique que le lexème associé au verbe infinitif a été étendu pour prendre en compte l'argument sujet du verbe causatif.

2.1.1 Distinction entre *faire_à* et *faire_{par}* - Procédure SpecifyRestructLex

Dans le processus de restructuration, l'analyseur associe toujours la deuxième position de la table d'argument du lexème lié au verbe infinitif aux caractéristiques du sujet profond du verbe infinitif. Ces caractéristiques permettent, entre autres, de distinguer les constructions *faire_à* et *faire_{par}*.

Une construction *faire_à* est identifiée si le deuxième argument du lexème a la valeur **internalized_subject** - noté **theta6** dans son champ **theta**. Pour reconnaître cette construction dans la suite du processus de transfert, le trait **vf2** est inclus dans le champ **v_features** du verbe infinitif.

Une construction *faire_{par}* est identifiée si le deuxième argument a la fonction grammaticale d'objet prépositionnel et si son champ **theta** a la valeur **agent**. Le verbe infinitif porte alors le trait **vf3**.

Certaines constructions causatives sont ambiguës du fait qu'elles peuvent être interprétées comme une construction *faire_à* tout comme une construction *faire_{par}*. Cette ambiguïté est levée par l'interaction. Quand l'interaction est inactive, la construction *faire_{par}* est adoptée par défaut. L'ambiguïté est véhiculée par une phrase telle que (6a) où le syntagme [PP à Marie] peut être assimilé comme sujet profond ou argument prépositionnel du verbe *écrire*.

(6)a. Jean fait écrire la lettre à Marie.

Si le syntagme [PP à Marie] est le sujet profond - [NP Marie] est l'auteur de la lettre, alors on est dans le cas d'une construction *faire_à*. La phrase cible est (7a). Pour une construction *faire_{par}*, par contre, le syntagme [PP à Marie] est l'argument prépositionnel - [NP Marie] est le destinataire de la lettre. Le sujet profond du verbe infinitif *écrire* n'est pas exprimé. La phrase cible devient (7b).

(7)a. Jean made Marie write the letter.

b. John had the letter written to Mary.

2.1.2 Détermination de la structure profonde du verbe infinitif cible

Lors de la recherche du lexème source qui représente la structure profonde source, on mettra en correspondance les arguments du lexème et les arguments du verbe infinitif - Procédure **MatchWithLexeme**. Il faut ensuite tenir compte de l'extension du lexème source pendant la restructuration pour établir la fonction grammaticale cible d'un argument source - Procédure **MapTargetLexeme**.

Procédure MatchWithLexeme	Procédure MapTargetLexeme
-----	-----
Lexème Source GOUTER	Argument Source
1. Sujet	[DP Marie]
2. Obj Prép + theta6	[PP à un ami]
3. Obj Direct	[DP le vin]
-----	-----
----- Informations du dictionnaire bilingue -----	-----

Ainsi dans notre exemple d'une construction *faire_a*, on trouvera le tableau ci-dessus qui exprime la fonction grammaticale cible d'un argument source.

La restructuration, rappelons-le, associe le sujet profond du verbe infinitif avec la deuxième position de la table d'argument du lexème et décale d'un cran tous les autres arguments du lexème. On peut, par conséquent, déduire la structure profonde du verbe infinitif. Dans notre exemple, on aura

Goûter (sujet profond [PP à un ami], objet direct [DP le vin])

Le dictionnaire bilingue permet ensuite d'établir que l'argument [PP à un ami] correspond au sujet cible et [DP le vin] à l'objet direct cible - Procédure **MapTargetLexeme**. Ces arguments sont transférés - Procédure **TrArguments** - pour construire la structure profonde cible du verbe infinitif.

Taste (sujet [DP a friend], objet direct [DP the wine])

Dans les autres exemples, on a schématiquement les informations suivantes :

Procédure MatchWithLexeme		Procédure MapTargetLexeme	
Lexème Source	Argument Source	Lexème Cible	
REPARER		FIX	
1. Sujet	[DP Jean]	1. Sujet	
2. Obj Prép + agent	[PP par un ami]	2. Obj Direct	
3. Obj Direct	[DP la voiture]		
----- Informations du dictionnaire bilingue -----			

Réparer (sujet profond [PP par un ami], Obj Direct [DP la voiture])

Fix (sujet [DP a friend], Obj. Direct [the car])

Procédure MatchWithLexeme		Procédure MapTargetLexeme	
Lexème Source	Argument Source	Lexème Cible	
DORMIR		SLEEP	
1. Sujet	[DP Paul]	1. Sujet	
2. Obj Direct + theta6	[DP les enfants]		
----- Informations du dictionnaire bilingue -----			

Dormir (sujet profond [PP les enfants])

Sleep (sujet [DP the children])

